Chapter 9
Reconfigurable Guidance System

This chapter presents a guidance algorithm for a UAV. It combines a nonlinear lateral guidance control law, originally designed for UAVs tracking circles for mid-air rendezvous, with a new simple adaptive path-planning algorithm. Preflight path planning consists only of storing a few waypoints guiding the aircraft to its targets. The chapter presents an efficient way to model no-fly zones (NFZ), to generate a path in real time to avoid known or "pop-up" obstacles, and to reconfigure the flight path in the event of reduced aircraft performance. Simulation results show the good performance of this reconfigurable guidance system which, moreover, is computationally efficient [1, 2].

9.1 Introduction

Over the last two decades, many path-planning algorithms have been investigated, especially for ground robots, for a single UAV, and more recently for a formation of UAVs. Among the methods used in path planning, we can mention the PRM method [3], which explores all the possible paths within the space surrounding the vehicle and finally selects the lowest cost route. However, the computational load makes the PRM method impractical for real-time path planning in small UAVs. An extension to the PRM method has recently been presented in [4]. It is called modified *rapidly-exploring random trees*, which is capable of efficiently searching for feasible paths in the space while taking into account constraints from the vehicle performance. However, efforts are still going on to implement an on-the-fly path-replanning system as pop-up obstacles are discovered or when the performance of the vehicle degrades.

There are other methods based on *potential field* functions. However, the primitive forms of potential field functions present some difficulties when choosing an appropriate potential function, and the algorithm may be stuck

at some local minimum [5]. Since then, a whole family of potential field methods with superior performance has been developed. They are known as *navigation functions* [6, 7]. Other path-planning techniques are based on optimization methods, such as *mixed integer linear programming* or MPC techniques [8], which still involve intensive computations.

In this chapter, we present a reconfigurable guidance algorithm for a UAV. It newly combines the lateral guidance control law from [9] and [10], originally designed for UAVs tracking circles for mid-air rendezvous, with a new, simple adaptive path-planning algorithm, which takes advantage of the curve path-following property of the above-mentioned lateral guidance law. This path-planning method generates on-line a flight path based on predefined waypoints, takes into account the aircraft performance, avoids known or appearing obstacles, is simple to implement, and requires low computational power.

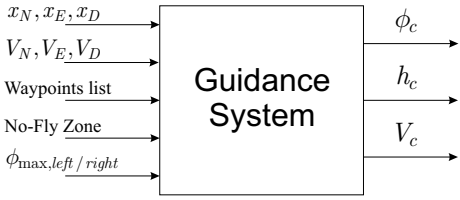


Fig. 9.1 Guidance system inputs and outputs

As shown in Fig. 9.1, the guidance system needs six inputs. The first input concerns the aircraft's current ground position ($x_N,\ x_E,\ x_D$). The second input is the aircraft's ground velocity ($V_N,\ V_E,\ V_D$). The mission of the aircraft is defined by a list of waypoints through which the aircraft is to fly. Furthermore, if in the area of the flight operation some obstacles or NFZ are known in advance or appear during the flight, their location and dimensions can be specified to the guidance system via the fourth input. A constraint on the maximum bank angle $\phi_{max,left/right}$ is given to the guidance system. Finally, the parameter τ_{roll} is provided as an estimate of the maximum time needed to bank the aircraft to ϕ_{max}. Note that the last four inputs can be changed dynamically, and the first two inputs are obviously constantly updated.

The outputs of the guidance system are the commanded bank angle ϕ_c, whose value is computed by the lateral guidance system detailed in Sects. 9.2 and 9.3. The altitude command signal h_c is computed by the altitude guidance system described in Sect. 9.4. Finally, the commanded aircraft velocity V_c can be adaptively controlled by the guidance system in order to efficiently avoid obstacles and reach the goals of the mission optimally. Note, however, that in this chapter the velocity command V_c is kept to a constant value.

9.2 Lateral Guidance System

9.2.1 Lateral Guidance Control Law for Trajectory Tracking

Consider Fig. 9.2, where an aircraft has to be guided to track the desired path. From the current location of the aircraft O, we can draw a circular arc that intersects the desired path at a "reference point" P, where R is the radius of the circle-arc OP, L_1 is the segment that joins the center of the aircraft O to the reference point P, and η is the angle between the aircraft's velocity vector and the line L_1.

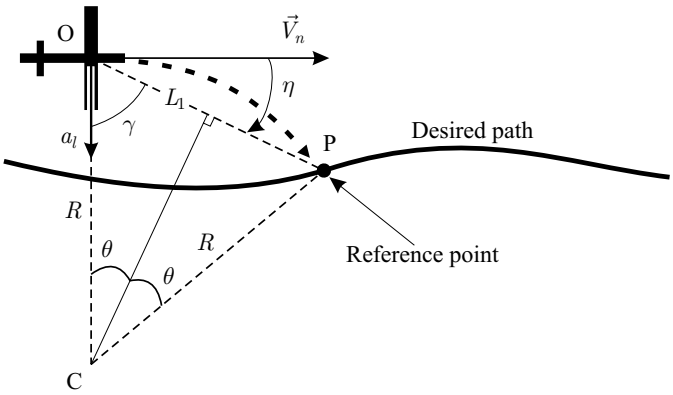


Fig. 9.2 Guidance law geometry

The lateral acceleration required to bring the aircraft to the reference point following the arc of a circle is

$$a_l = \frac{V_n^2}{R} \,, \tag{9.1}$$

where the ground speed of the aircraft (taken in the local navigation frame) is $V_n = \sqrt{V_N^2 + V_E^2}$. Let us express R in terms of the distance L_1 and the angle η. The triangle (OCP) is isosceles in C, therefore, we have $L_1 = 2R\sin\theta$, or also $L_1 = 2R\cos\gamma$. Moreover, the angle $\gamma = \frac{\pi}{2} - \eta$, and consequently, the length L_1 can be expressed in terms of the angle η as follows:

$$L_1 = 2R\sin\eta \,, \tag{9.2}$$

$$\iff R = \frac{L_1}{2\sin\eta} \,. \tag{9.3}$$

The lateral acceleration in (9.1) can now be written as

$$a_l = \frac{2V_n^2}{L_1} \sin \eta \ . \tag{9.4}$$

In turn, the lateral acceleration a_l is converted to a bank angle command, $\phi_{com} \approx a_l/g$ (see Appendix D.2).

9.2.2 Advantages and Properties of the Method

9.2.2.1 Remarkable Performance for Curve Path Following

This control law is remarkable in the sense that it is particularly suited to fly circles. Indeed, if the aircraft is following a desired circular path, then the acceleration command a_l generated by the guidance system is exactly the same as the associated centripetal acceleration. In other words, the guidance logic chooses a reference point on the desired path at a distance L_1 ahead of the aircraft, and generates the acceleration command that would lead the vehicle to hit the point after flying a circular arc, thus leading to zero steady-state error for a circular path. As shown in Chap. 3 of [9], the performance of such a lateral guidance law for circle following in the presence of wind is superior to that obtained with PD or PID controllers. With the nonlinear guidance law $a_l = \frac{2V_n^2}{L_1} \sin \eta$, the vehicle ground speed V_n is used to generate the acceleration command, which intrinsically takes into account the inertial velocity changes due to the wind effects, and adapts to the situation accordingly.

9.2.2.2 Properties Associated with the Angle η

Equation 9.4 shows that the direction of the acceleration a_l depends on the sign of η. For example, if the reference point is on the right side of the direction of the aircraft velocity vector, then the aircraft will be commanded to accelerate to the right, and finally the aircraft will tend to align its velocity vector with the direction of $\boldsymbol{L_1}$.

In practice the distance L_1 is fixed to a certain value, and two cases arise:

- If the aircraft is far away from the desired path, then the direction of $\boldsymbol{L_1}$ makes a large angle with the desired path. Therefore, the guidance law chooses the reference point on the desired trajectory in such a way that the aircraft rotates its velocity direction to approach the desired path at a large angle.
- If the aircraft is close to the desired path, then the direction of L_1 makes a small angle with the desired path. Therefore, the guidance law chooses the reference point on the desired trajectory in such a way that the aircraft rotates its velocity direction to approach the desired path at a small angle.

Since the angle η contains the information about the upcoming path, this geometric factor has the effect of a feedforward control.

9.2.3 Drawback of the Method

A drawback of the control law in (9.4) is that it does not contain any element of an integral control. Therefore, the lateral guidance law requires to be provided with an unbiased lateral acceleration measurement, and a non-biased bank angle estimate.

9.2.4 Selection of L_1

The design parameter in the lateral guidance logic is the distance L_1 between the vehicle and the reference point as shown in Fig. 9.3. It is explained in [9] that for a small magnitude of η, the guidance formula can be approximated in terms of the cross-track error y as follows:

$$a_l = \frac{2V_n^2}{L_1} \sin\eta \approx 2\frac{V_n}{L_1}\left(\dot{y} + \frac{V_n}{L_1} y\right) \ . \tag{9.5}$$

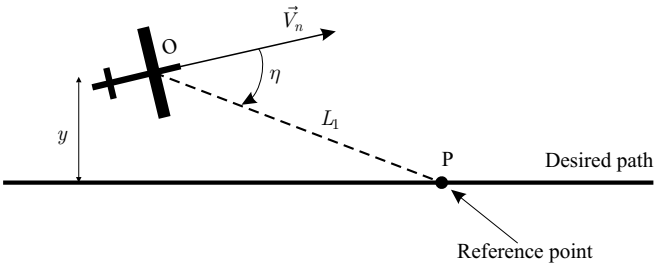


Fig. 9.3 Guidance law geometry

Equation 9.5 shows that the guidance law is equivalent to a PD controller, in which the ratio between the vehicle speed V_n and the distance L_1 is an important factor that behaves as the gain of the controller. A small value of L_1 leads to a high control gain and *vice versa*. The control gain is limited by the inner loop bank control bandwidth (2−3 rad/s). With a nominal flight velocity of around 25 m/s, the distance L_1 has been chosen to be $L_1 = 150\,\text{m}$.

9.2.5 Path-planning Objective

The main objective of this chapter is to generate on-line an appropriate reference path from which the reference point P can be selected and used by the lateral guidance control law.

9.3 Regular Waypoint Tracking

The regular waypoint tracking algorithm guides the aircraft through the predefined waypoints. An imaginary segment joins two consecutive points, and we define the *segment* k to be the segment that joins the waypoints WP_k and WP_{k+1}. The reference point, P, from the guidance law presented above, lies on this segment and is L_1 distant from the center of the aircraft; see Fig. 9.4.

9.3.1 Computation of the Reference Point P

The angle of the segment k with respect to North is defined as follows:

$$\psi_{seg(k)} = \tan^{-1}\left(\frac{WP_{k+1,E} - WP_{k,E}}{WP_{k+1,N} - WP_{k,N}}\right) \in [-\pi;\pi]\,. \tag{9.6}$$

The coordinates of the current location of the center of the aircraft are in the North−East plane (X_N, X_E). The angles χ and λ, and the distance d_1 as shown in Fig. 9.4 can be computed as follows:

$$\begin{aligned}
\chi &= \tan^{-1}\left(\frac{X_E - WP_{k,E}}{X_N - WP_{k,N}}\right) \in [-\pi;\pi]\,,\\
\lambda &= |\psi_{seg(k)}| - |\chi|\,,\\
d_1 &= \sqrt{(X_E - WP_{k,E})^2 + (X_N - WP_{k,N})^2}\,.
\end{aligned} \tag{9.7}$$

The distance between WP_k and the reference point P is given by

$$[WP_k, P] = [WP_k, H] + [H, P] = d_1\cos\lambda + \sqrt{L_1^2 - {d_1}^2\sin^2\lambda}\,. \tag{9.8}$$

Finally, the coordinates of the reference point P can be computed with

$$\begin{aligned}
P_N &= WP_{k,N} + [WP_k, P]\cos\psi_{seg(k)}\,,\\
P_E &= WP_{k,E} + [WP_k, P]\sin\psi_{seg(k)}\,.
\end{aligned} \tag{9.9}$$

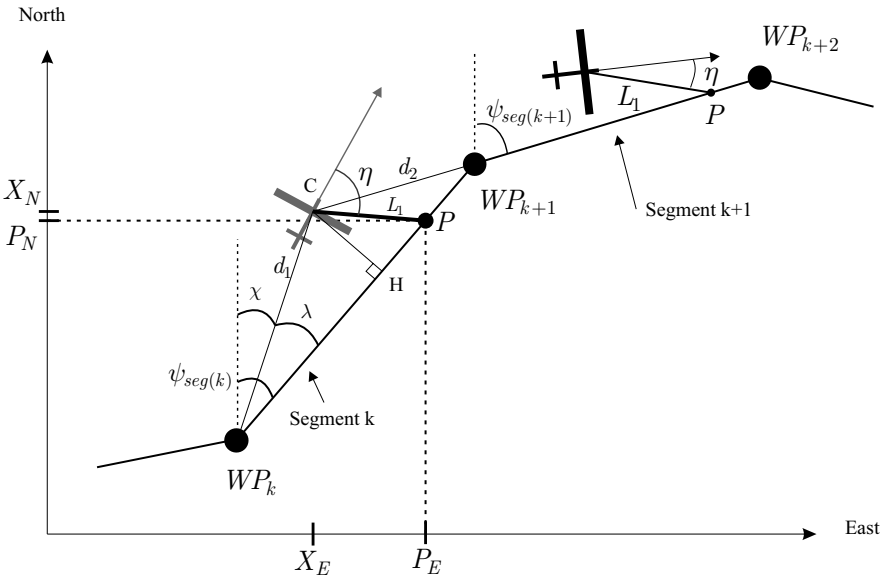


Fig. 9.4 Regular waypoint tracking

9.3.2 Logic for Segment Switching

As the aircraft flies, the reference point P also moves along the desired trajectory, which consists of consecutive segments. Therefore, the guidance algorithm has to select properly the current segment on which the reference point P is to be selected.

Equation 9.10 gives the two conditions that are continuously checked:

$$d_2 \geq L_1, \qquad d_1 \cos\lambda < \|WP_k, WP_{k+1}\| \; . \tag{9.10}$$

If one of the two conditions is not satisfied anymore, the guidance system has to select a reference point P on the next segment $[WP_{k+1}, WP_{k+2}]$. On the other hand, if the two conditions are still satisfied, the guidance system computes a reference point P on the segment $[WP_k, WP_{k+1}]$. In that case, the path-planning system has to further check the lateral distance between the aircraft and the segment followed in order to make sure that it selects a point on the current segment that is L_1 distant from the aircraft.

Two subcases can be distinguished:

- If $|\lambda| > \pi/2$, the aircraft is somewhere behind the first point of the current segment. In that case, the distance L_1 is selected as $L_1 = \max(L_1, d_1)$.
- If $|\lambda| < \pi/2$, the guidance algorithm proceeds in checking the lateral distance $[C, H] = d_1 \sin\lambda$. If $[C, H] > L_1$, then the distance L_1 is assigned a new value with $L_1 = 1.1 \times [C, H]$.

In this way, we always ensure that the arm L_1 is long enough to intersect the desired trajectory, so that a reference point P always exists.

9.3.3 Computation of the Roll Angle Command ϕ_{com}

The direction of the aircraft's ground speed is computed with

$$\psi = \tan^{-1}\left(\frac{V_E}{V_N}\right) \quad \in [-\pi; \pi] \; . \tag{9.11}$$

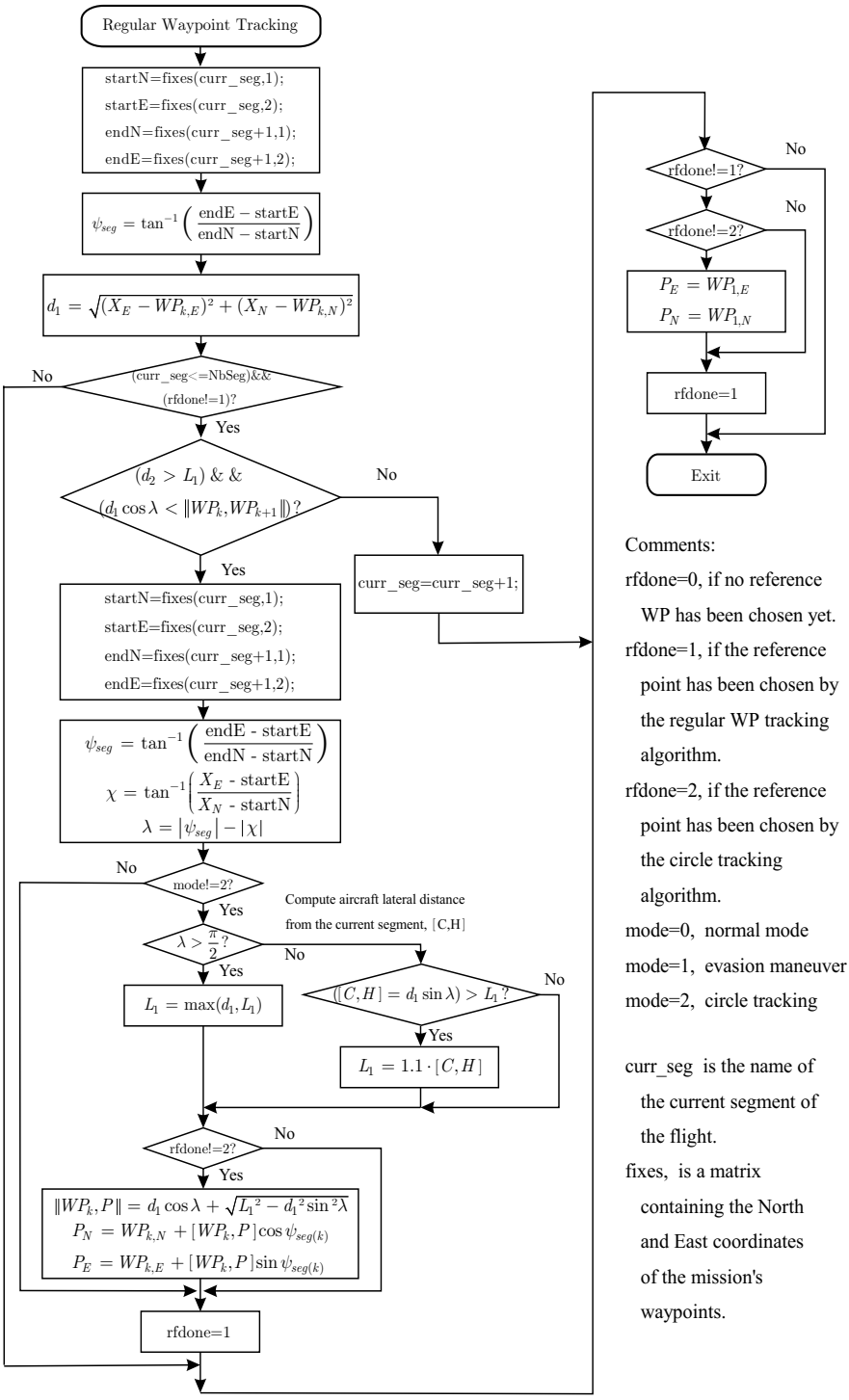
The lateral guidance control law needs the angle η, which is computed as follows (see Fig. 9.6):

$$\begin{aligned} \eta &= \Omega - \psi \; , \\ \Omega &= \tan^{-1}\left(\frac{P_E - X_E}{P_N - X_N}\right) \quad \in [-\pi; \pi] \; . \end{aligned} \tag{9.12}$$

In practice, we want the angle η to be in the range $[-\pi; \pi]$, therefore the following code implementation is used

$$\begin{aligned} &\text{while } (\eta > \pi) \;\; \eta = \eta - 2\pi \; , \\ &\text{while } (\eta < -\pi) \;\; \eta = \eta + 2\pi \; . \end{aligned} \tag{9.13}$$

Once the angle η is in the range $[-\pi; \pi]$, the angle η is further limited (saturated if needed) to be in the range $[-\pi/2; \pi/2]$. The reason for this is that when the angle η becomes large and approaches $\pi/2$ (or $-\pi/2$, respectively) we want the aircraft to bank to the right (or to the left, respectively) as much as is possible to quickly come closer again to the reference trajectory. Indeed, since the angle η is used in the roll angle command with $\phi_{com} = 2V_n^2 \sin\eta/(L_1 g)$, if η exceeded $\pi/2$ $(-\pi/2)$ in the range $[\pi/2; \pi]$ $([-\pi/2; -\pi])$ the term $\sin\eta$ would decrease in amplitude, which means that the commanded roll angle also decreases, and the aircraft would come back more slowly to the reference trajectory, which is not desired. Finally, the control signal for the roll angle ϕ_{com} is saturated within the range $[-\phi_{max}; \phi_{max}]$.



Comments:

rfdone=0, if no reference WP has been chosen yet.

rfdone=1, if the reference point has been chosen by the regular WP tracking algorithm.

rfdone=2, if the reference point has been chosen by the circle tracking algorithm.

mode=0, normal mode

mode=1, evasion maneuver

mode=2, circle tracking

curr_seg is the name of the current segment of the flight.

fixes, is a matrix containing the North and East coordinates of the mission's waypoints.

Fig. 9.5 Regular waypoint tracking

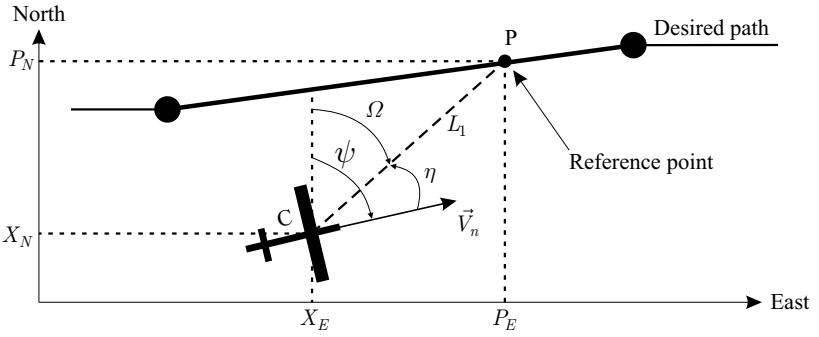


Fig. 9.6 Determination of the angle η

9.4 Altitude Guidance Law

The altitude control signal is h_c and is computed as follows:

$$h_c = WP_{k,D} + \gamma d_1 \cos\lambda \,, \tag{9.14}$$

with the desired flight path angle defined as follows:

$$\gamma = \tan^{-1}\left(\frac{WP_{k+1,D} - WP_{k,D}}{\sqrt{(WP_{k+1,N} - WP_{k,N})^2 + (WP_{k+1,E} - WP_{k,E})^2}}\right) \,, \tag{9.15}$$

with λ and d_1 defined in (9.7).

Figure 9.7 shows the aircraft flying from waypoint WP_k to WP_{k+1}. The desired altitude is h_c computed as a function of the aircraft's ground position. The aircraft's current altitude is X_D, which is obviously too high compared with h_c in Fig. 9.7, and the altitude controller of Section 7.5 must correct this altitude error.

Remarks:

- Equations 9.14 and 9.15 are used to generate the altitude reference signal in the normal waypoint tracking mode.
- In the modes *evasion maneuver* (mode 1) or *circle tracking* (mode 2), the altitude reference signal is kept at the value where it was before the guidance system entered one of these modes. Indeed, in mode 1 or 2 the aircraft takes a path that is not known in advance and for which no altitude may be specified. Once the guidance system switches back to *normal waypoint*

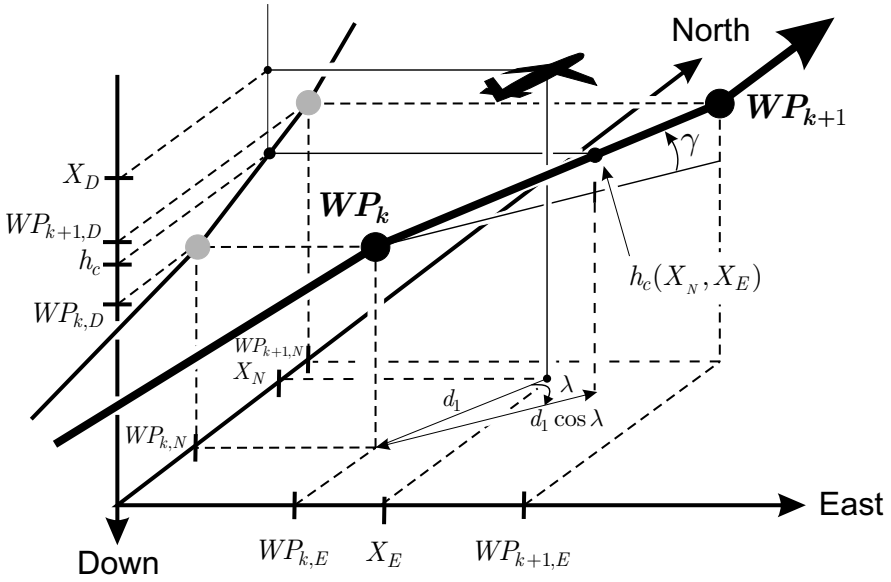


Fig. 9.7 Altitude tracking geometry

tracking mode (mode 0), the altitude control signal is again constantly recomputed.

9.5 NFZ and Obstacles

Before the flight, the location of any known NFZ are stored in the memory of the autopilot. If the UAV is equipped with scanning sensors that can detect pop-up obstacles, our path-planning system will recompute on the fly a new trajectory. It determines whether an NFZ or an obstacle interferes with the planned path by using an imaginary "detection line" of length R_{LA} in front of the aircraft, as shown in Fig. 9.8. The distance R_{LA} defines the so-called "look-ahead distance". If any part of this detection line penetrates an NFZ or an obstacle, avoidance action (mode 1) is immediately taken as described in the next section.

9.5.1 Definition of an NFZ

An NFZ is any airspace in which an aircraft is not permitted to fly. This airspace can be of any arbitrary shape. However, in order to simplify the

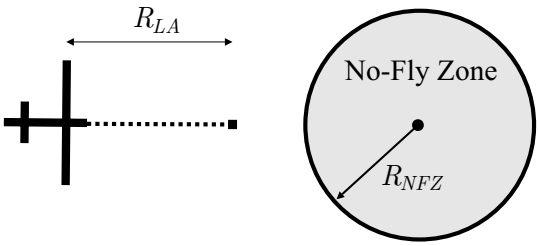


Fig. 9.8 NFZ and look-ahead distance R_{LA}

guidance algorithm, two conditions are imposed on how the NFZ is represented.

First, the vertical limits of the NFZ are not considered such that the NFZ is essentially a 2D surface. The aircraft is not allowed to pass over the NFZ. Second, the shape of the NFZ is chosen to be a circle. In this way, the avoidance maneuver can be an arc of a circle in order to benefit from the guidance control law especially suited to track circles, which are described by only two parameters, namely their center and their radius.

Although this chapter discusses the avoidance of one circular NFZ only, the algorithm can be extended to multiple NFZs with some simple modifications. Also, a complex NFZ shape can be represented by multiple circles.

9.5.2 Choice of an Appropriate Look-ahead Distance R_{LA}

The distance R_{LA} defines the so-called "look-ahead distance". If any part of this detection line penetrates an NFZ or an obstacle, avoidance action is immediately taken as described in the next section.

The look-ahead distance R_{LA} is chosen such that the aircraft will fly an arc that stays just outside the NFZ and start the evasion maneuver as late as possible. The value of R_{LA} depends on the radius R_{NFZ} of the NFZ, the ground speed of the aircraft V_n, and the maximum bank angle of the aircraft ϕ_{max}. Given these parameters, and assuming a coordinated turn, the minimum turn radius that the aircraft can fly is given by

$$R_{min} = \frac{V_n^2}{g \tan(\phi_{max})} ,$$
$$V_n = \sqrt{V_N^2 + V_E^2} . \tag{9.16}$$

The subscript n attached to the ground speed of the aircraft V_n indicates that the speed is taken in the local navigation frame.

In the case of an NFZ with infinite radius, the aircraft would have to make a 90° turn, in which case $R_{LA,min} = R_{min}$. For any NFZ with a finite radius, the aircraft has to turn less than 90° to avoid it. Figure 9.9 shows the situation where the aircraft is at the point where it begins its turn and is guided so that its path becomes tangent to the edge of the NFZ. A triangle can be set up with vertices at the center of the NFZ, at the center of the aircraft, and at a point R_{min} off the right wing-tip. The minimum look-ahead distance $R_{LA,min}$ is constructed using the Pythagorean theorem

$$(R_{LA,min} + R_{NFZ})^2 + R_{min}^2 = (R_{min} + R_{NFZ})^2 \, . \tag{9.17}$$

By expanding both sides and regrouping the terms, we obtain the following equation, where the distance $R_{LA,min}$ is the unknown variable

$$R_{LA,min}^2 + 2R_{LA,min}R_{NFZ} - 2R_{min}R_{NFZ} = 0 \, . \tag{9.18}$$

Equation 9.18 admits two real solutions of which we only keep the positive one

$$R_{LA,min} = \sqrt{R_{NFZ}}\sqrt{R_{NFZ} + 2R_{min}} - R_{NFZ} \, . \tag{9.19}$$

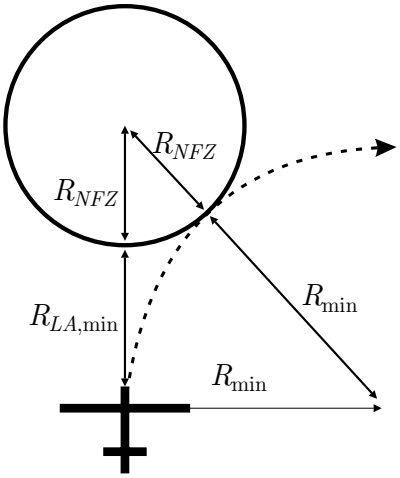


Fig. 9.9 Approaching an NFZ or an obstacle. Definition of the look-ahead distance R_{LA}

To obtain the final value for R_{LA}, compensation must be made for the delay needed to initiate the turn and to bank to ϕ_{max}. The assumption is made that while the aircraft is initiating the turn, it continues to fly level, and then as soon as it reaches ϕ_{max} it takes a minimum radius turn. The characteristic time τ_{roll} to roll to ϕ_{max} can be multiplied by the aircraft's speed to obtain the distance the aircraft will travel during this delay, which

is added to $R_{LA,min}$. The resulting look-ahead distance is

$$R_{LA} = R_{LA,min} + V_n \, \tau_{roll} \; . \tag{9.20}$$

9.6 Detection of the NFZ

As mentioned before, the algorithm monitors an "imaginary" line of length R_{La} ahead of the aircraft, and checks if it penetrates any NFZ or obstacle. First, the distance D_{NFZ} from the aircraft to the center of the NFZ is calculated; see Fig. 9.10:

$$D_{NFZ} = \sqrt{(NFZ_N - X_N)^2 + (NFZ_E - X_E)^2} \; . \tag{9.21}$$

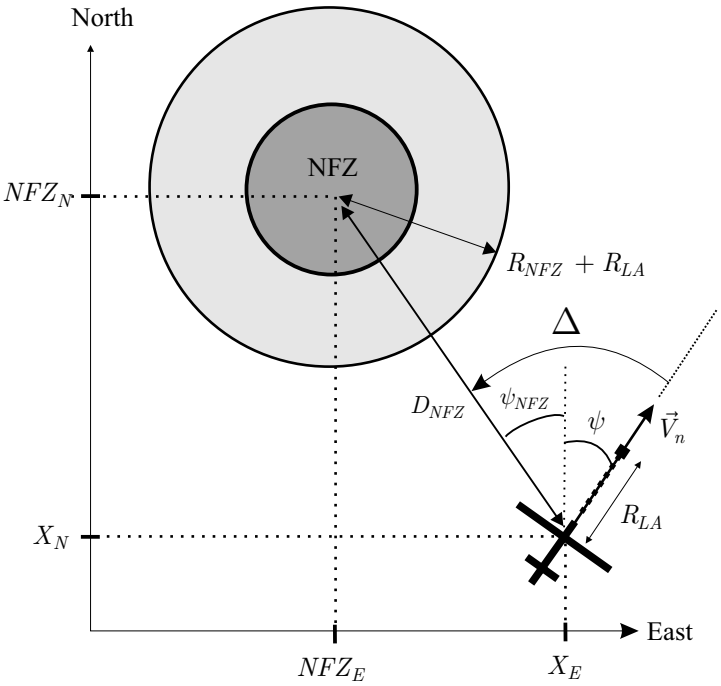


Fig. 9.10 Geometry of the NFZ approach

If the condition

$$D_{NFZ} \leq R_{NFZ} + R_{LA} \tag{9.22}$$

is satisfied, where the length R_{NFZ} is the radius of the NFZ, then the aircraft is considered to be within range of the NFZ, which means that the aircraft

is in the light-gray area in Fig. 9.10. In this case, a further check is made to see if a part of the detection line is touching the NFZ.

There are two possible cases shown in Figs. 9.11 and 9.12 depending on the orientation of the aircraft. The distances D_{NFZ} and R_{LA} are known. We also define the angle ψ_{NFZ} between the North direction at the center of the aircraft and the segment of length D_{NFZ} as follows:

$$\psi_{NFZ} = \tan^{-1}\left(\frac{NFZ_E - X_E}{NFZ_N - X_N}\right) . \tag{9.23}$$

A pair of triangles is created as shown in Fig. 9.11. The length of the edges y and a can easily be calculated, using

$$\begin{aligned} \Delta &= |\psi_{NFZ} - \psi| , \ \in [-\pi; \pi] , \\ y &= D_{NFZ} \cdot \sin \Delta , \\ a &= D_{NFZ} \cdot \cos \Delta . \end{aligned} \tag{9.24}$$

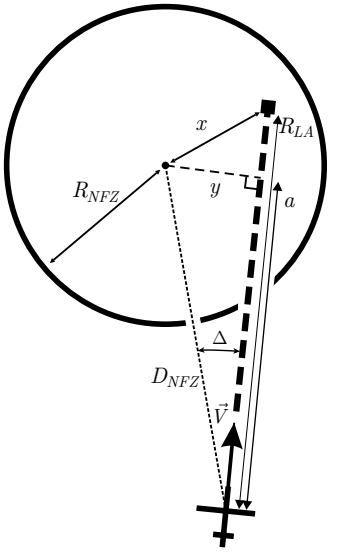


Fig. 9.11 Diagram of NFZ detection algorithm, Case 1 (detected)

Case 1: $a \leq R_{LA}$

Case 1 applies if $a \leq R_{LA}$. The limiting case occurs when edge a is tangent to the NFZ. Therefore, y will have a length equal to R_{NFZ}. Thus, the NFZ touches the detection line if $y \leq R_{NFZ}$.

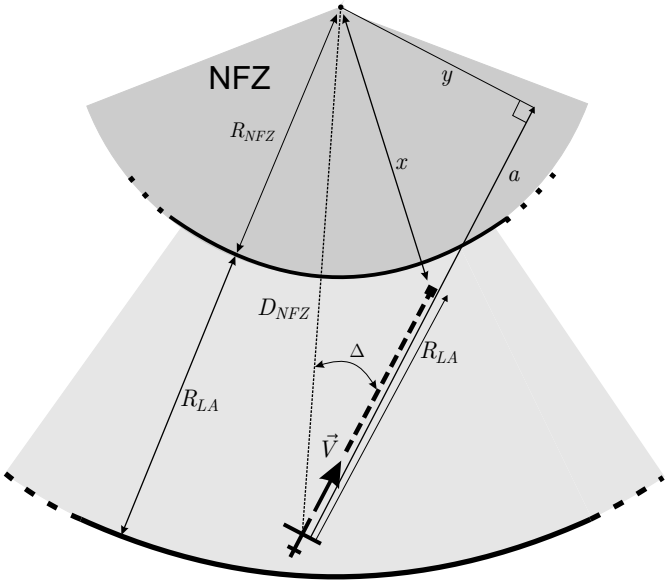


Fig. 9.12 Diagram of NFZ detection algorithm, Case 2 (not detected)

Case 2: $a > R_{LA}$

Case 2 applies if $a > R_{LA}$. The limiting case occurs when the end of the detection line is on the edge of the NFZ. This can be checked by comparing the length of edge x with the radius of the NFZ, so that the NFZ touches the detection line if $x \leq R_{NFZ}$, where $x = \sqrt{y^2 + (a - R_{LA})^2}$.

The check for Case 1 or Case 2 is only done if Δ is less than or equal to 90°. If Δ is greater than 90°, then the center of the NFZ lies behind the aircraft and no action is taken. Figure 9.13 shows the diagram of the NFZ detection algorithm.

9.7 NFZ Avoidance Algorithm

The NFZ avoidance algorithm guides the aircraft around any NFZ that the aircraft encounters. The avoidance method is designed to be simple to implement while allowing the aircraft to reach waypoints close to the edge of the NFZ.

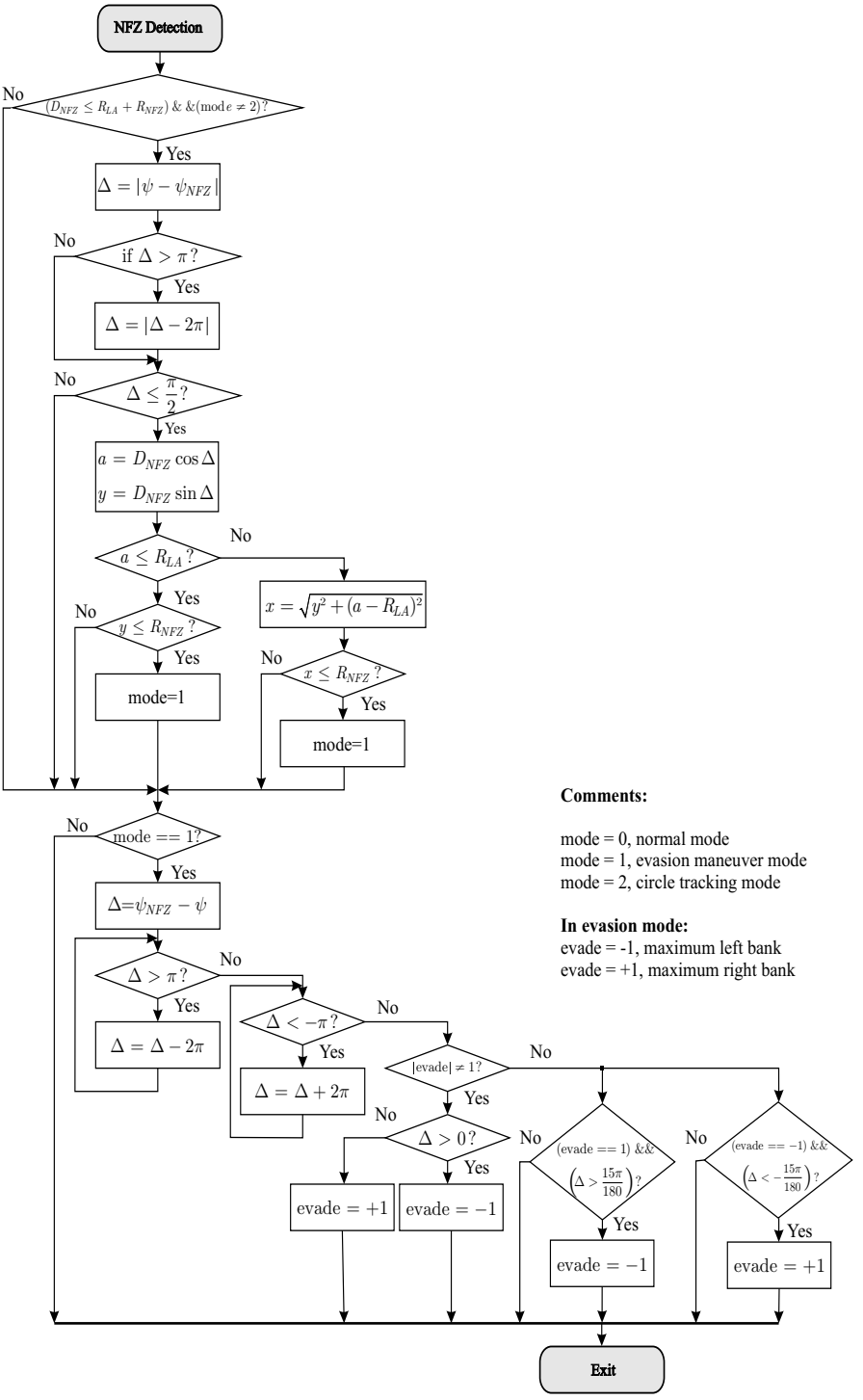


Fig. 9.13 Diagram of NFZ detection algorithm

9.7.1 On-line Selection of an Avoidance Path Template

One key feature of this avoidance method is the on-line generation of a circular arc around the NFZ as a reference path, drawn as a dashed line in Fig. 9.14. Such a path minimizes the distance the aircraft flies to avoid the NFZ. Moreover, we saw at the beginning of this chapter that the lateral guidance control law is particularly efficient in tracking circles.

Furthermore, choosing the reference path to be circular allows the template path to be easily defined in relationship to the NFZ dimensions. It is indeed defined by the center of the NFZ and a path radius, R_1, which is simply the NFZ radius plus a safety margin. The aircraft follows this path until it is able to continue towards the next waypoint in a straight line and without passing through the NFZ.

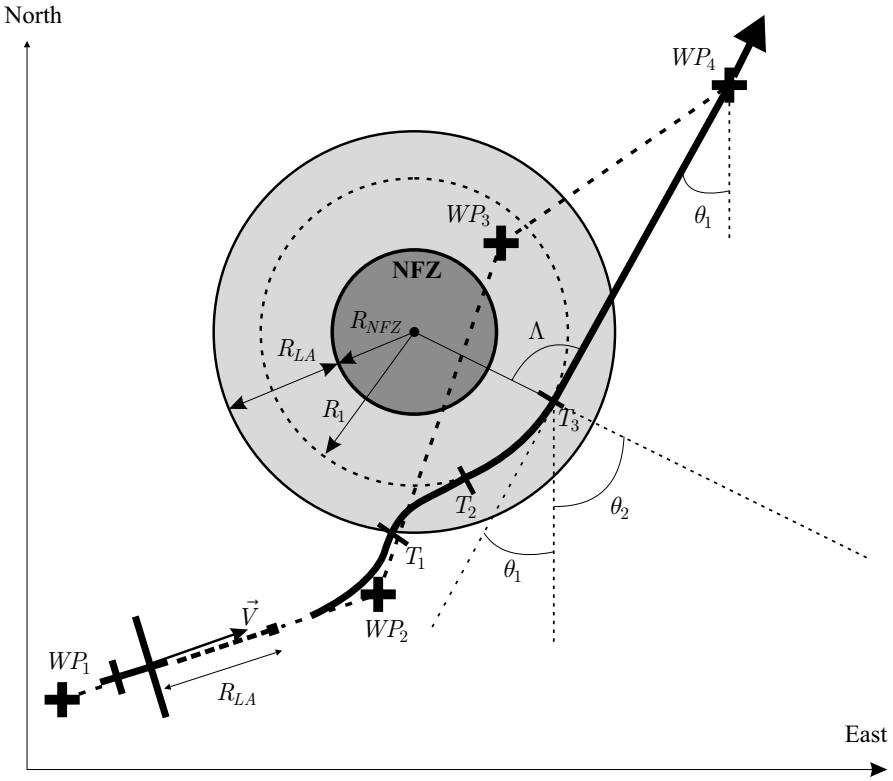


Fig. 9.14 Circular template path, waypoint tracking and reconfiguration

9.7.2 Entering the Circular Path Template

As soon as the obstacle is detected to be an immediate threat to the aircraft at point T_1, the guidance system begins an evasion maneuver with the minimum turn radius possible to reach tangentially the template path at point $T2$; see Fig. 9.14.

9.7.3 Choice of the Avoidance Side

Whether the guidance algorithm chooses to go left or right around the NFZ is determined by which side of the NFZ center the aircraft is already flying towards. If the aircraft's velocity vector is pointing to the right of the NFZ center, then the aircraft will fly around the NFZ on the right-hand side. If the velocity vector is pointing to the left-hand side, then the aircraft flies around the NFZ on the left-hand side. A circular NFZ makes this decision easy.

In practice, we compute $\Delta = \psi_{NFZ} - \psi \quad (\in [-\pi; \pi])$, and if $\Delta > 0$ then the aircraft flies on the left-hand side of the NFZ, otherwise it evades on the right-hand side; see Fig. 9.15.

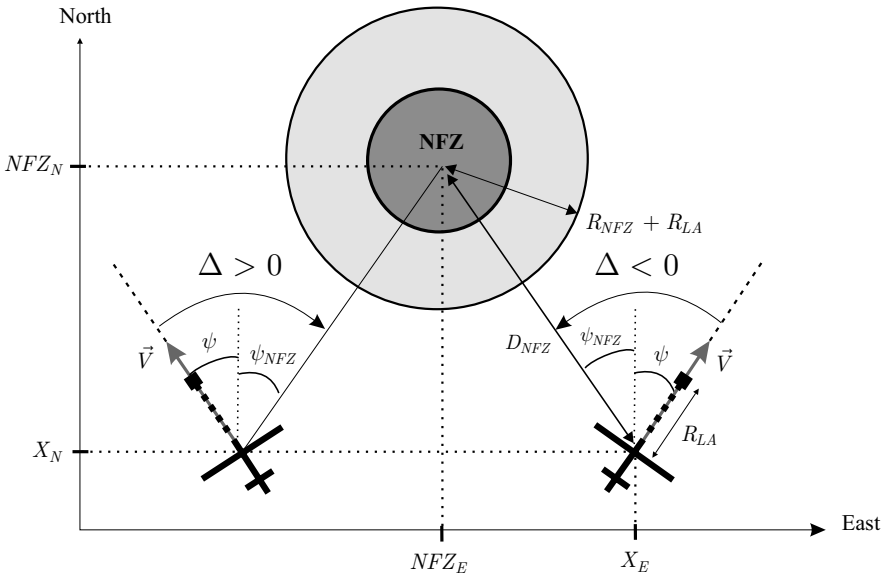


Fig. 9.15 NFZ evasion side

9.7.4 Generating the Template Path

Once the evasion maneuver is complete, the extremity of the monitoring line lies outside the NFZ. If the next waypoint to reach is obstructed by the NFZ, the guidance system guides the aircraft around the NFZ until there is a clear line of sight to the next valid waypoint. In this case, the guidance system has

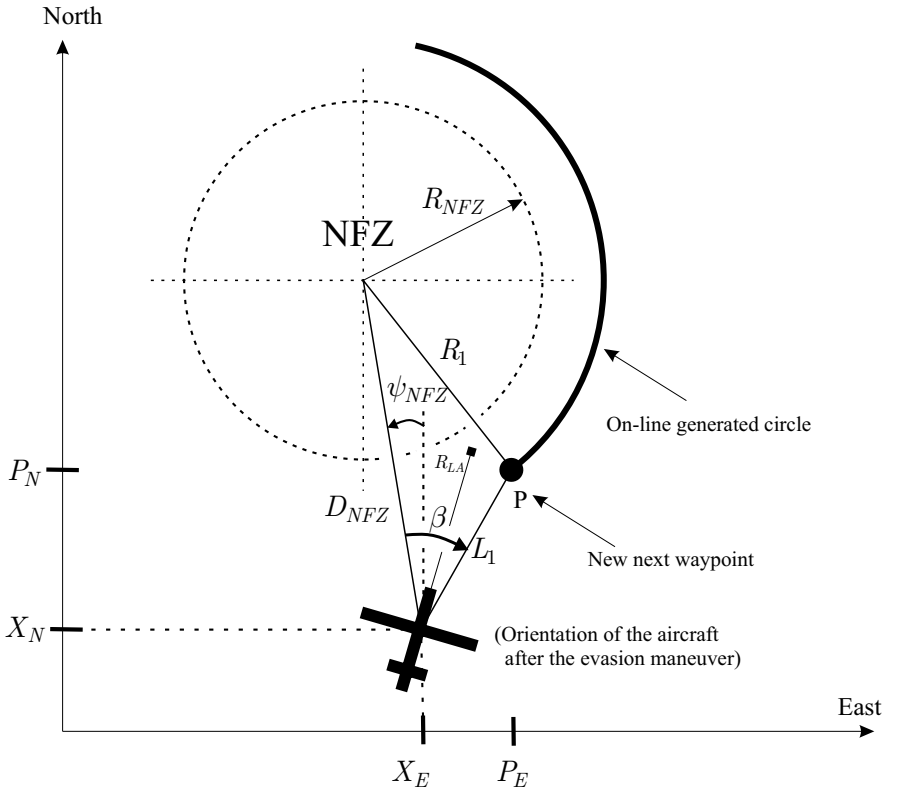


Fig. 9.16 Circular template path generation

to guide the aircraft to a point P that is on the circle R_1 and at a distance L_1 ahead of the aircraft. Using the law of cosines, the angle β (Fig. 9.16) between the segment that joins the center of the aircraft to the center of the NFZ and the segment from the center of the aircraft to the new reference point P is expressed as

$$\beta = \arccos\left(\frac{{D_{NFZ}}^2 + {L_1}^2 - {R_1}^2}{2D_{NFZ}\,L_1}\right) . \tag{9.25}$$

In order to make sure that the generated circle to avoid the NFZ is a feasible path for the aircraft, the radius R_1 is selected as

$$R_1 = \max\left(R_{min}, R_{NFZ} + \text{safety value}\right) . \tag{9.26}$$

The coordinates of the reference point P can then be computed as follows:

- If the NFZ is to be avoided on the right-hand side, we compute $\Psi_{avoid} = \psi_{NFZ} + \beta$
- If the NFZ is to be avoided on the left-hand side, we compute $\Psi_{avoid} = \psi_{NFZ} - \beta$

and finally

$$\begin{aligned} P_N &= X_N + L_1 \cos(\Psi_{avoid}) , \\ P_E &= X_E + L_1 \sin(\Psi_{avoid}) . \end{aligned} \tag{9.27}$$

9.7.5 Leaving the Circular Path Template

The aircraft follows the circular path until it is able to continue towards the next waypoint in a straight line and without passing through the NFZ. The point at which the guidance algorithm transitions back to normal guidance (mode 0) towards the next waypoint is T_3 in Fig. 9.14, and it occurs when there is a clear line of sight from the aircraft's current position to the next waypoint.

The next waypoint has to be reachable, which means that it should lie outside a no-fly area. Therefore, while initiating the evasion maneuver, the guidance system analyzes the waypoint list and checks if the waypoint WP_{k+1} that was tracked before the evasion maneuver is still a reachable point. The following condition must hold:

$$d_{NFZ/WP_{k+1}} = \sqrt{(NFZ_N - WP_{k+1,N})^2 + (NFZ_E - WP_{k+1,E})^2} > R_1 . \tag{9.28}$$

If it turns out that this waypoint is not reachable, the guidance system selects the next waypoint in the list, and so on. Figure 9.14 illustrates the waypoint reconfiguration. The aircraft makes a left turn at waypoint WP2 in the direction of WP3. Then the NFZ is detected and the guidance system guides the aircraft around the obstacle. The unreachable waypoint WP3 is discarded and instead WP4 becomes the next target.

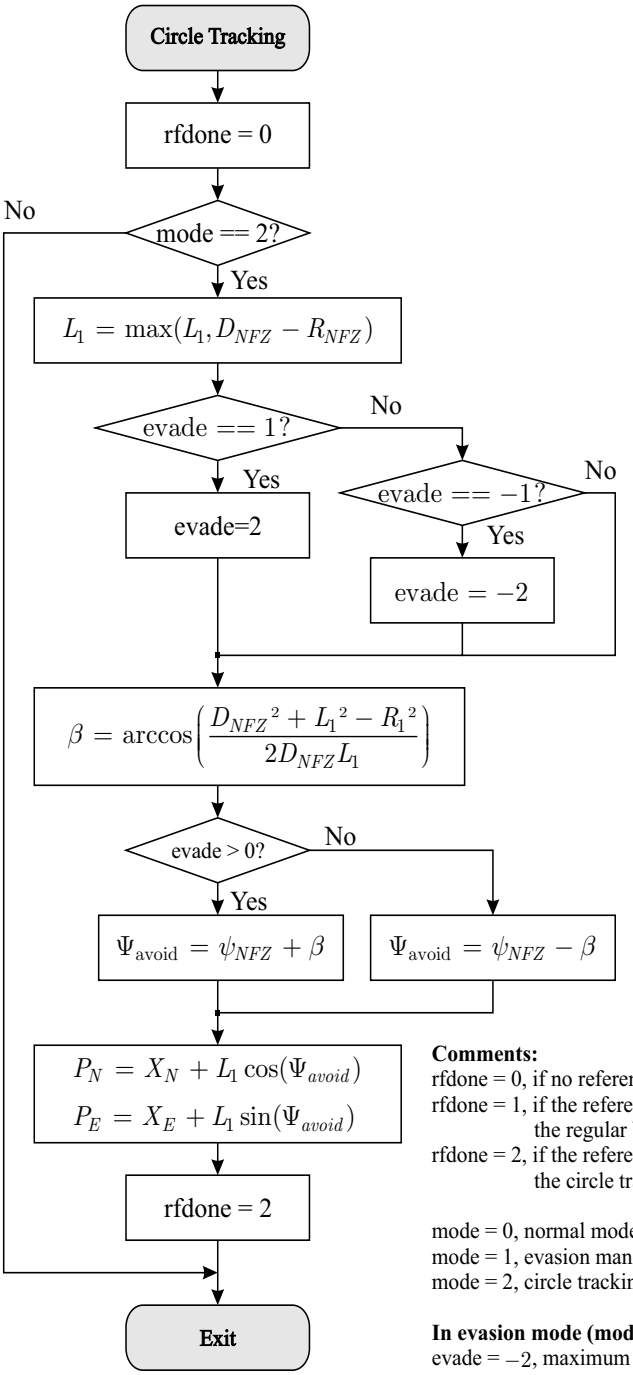


Fig. 9.17 Template path generation diagram

When should the guidance transition T_3 shown in Fig. 9.14 occur? A simple criterion is to monitor the angle Λ as shown in Fig. 9.14, which is the angle between the segment made by the aircraft's center and the next valid waypoint, and the segment made by the aircraft's center and the center of the NFZ. As soon as $|\Lambda| > \pi/2$, then there is a clear line of sight to the next valid waypoint. The angle Λ can be expressed simply as follows:

$$\Lambda = \theta_1 - \theta_2 \quad \in [-\pi; \pi] \,, \tag{9.29}$$

with

$$\begin{aligned} \theta_1 &= \tan^{-1}\left(\frac{WP_{k+1,E} - X_E}{WP_{k+1,N} - X_N}\right) \,, \\ \theta_2 &= \psi_{NFZ} = \tan^{-1}\left(\frac{NFZ_E - X_E}{NFZ_N - X_N}\right) \,. \end{aligned} \tag{9.30}$$

The decision steps for the next valid waypoint are summarized in Fig. 9.18.

9.7.6 Properties of the Guidance Schedule

The guidance schedule presented here has several desirable properties. It attempts to minimize the number of waypoints that are unreachable by initiating the avoidance maneuver as late as possible. It does not require any complex logic to decide how to avoid the NFZ. Finally, it minimizes the distance and time needed to return to the original flight path. It does this by flying directly to the next waypoint as soon as is safely possible.

9.8 Simulation

9.8.1 Simulation Set-up

Simulations were done using the nonlinear six degree-of-freedom computer model of a radio controlled aerobatic aircraft described in Chap. 3. The flight controllers used are described in Chaps. 7 and 8. The airspeed, the altitude, and the sideslip angle are kept constant.

Three similar scenarios were simulated, with the results presented below. In all scenarios, the aircraft is following a desired path that passes through an NFZ. The simulation was done with a maximum bank angle of $\phi_{max} = 30°$.

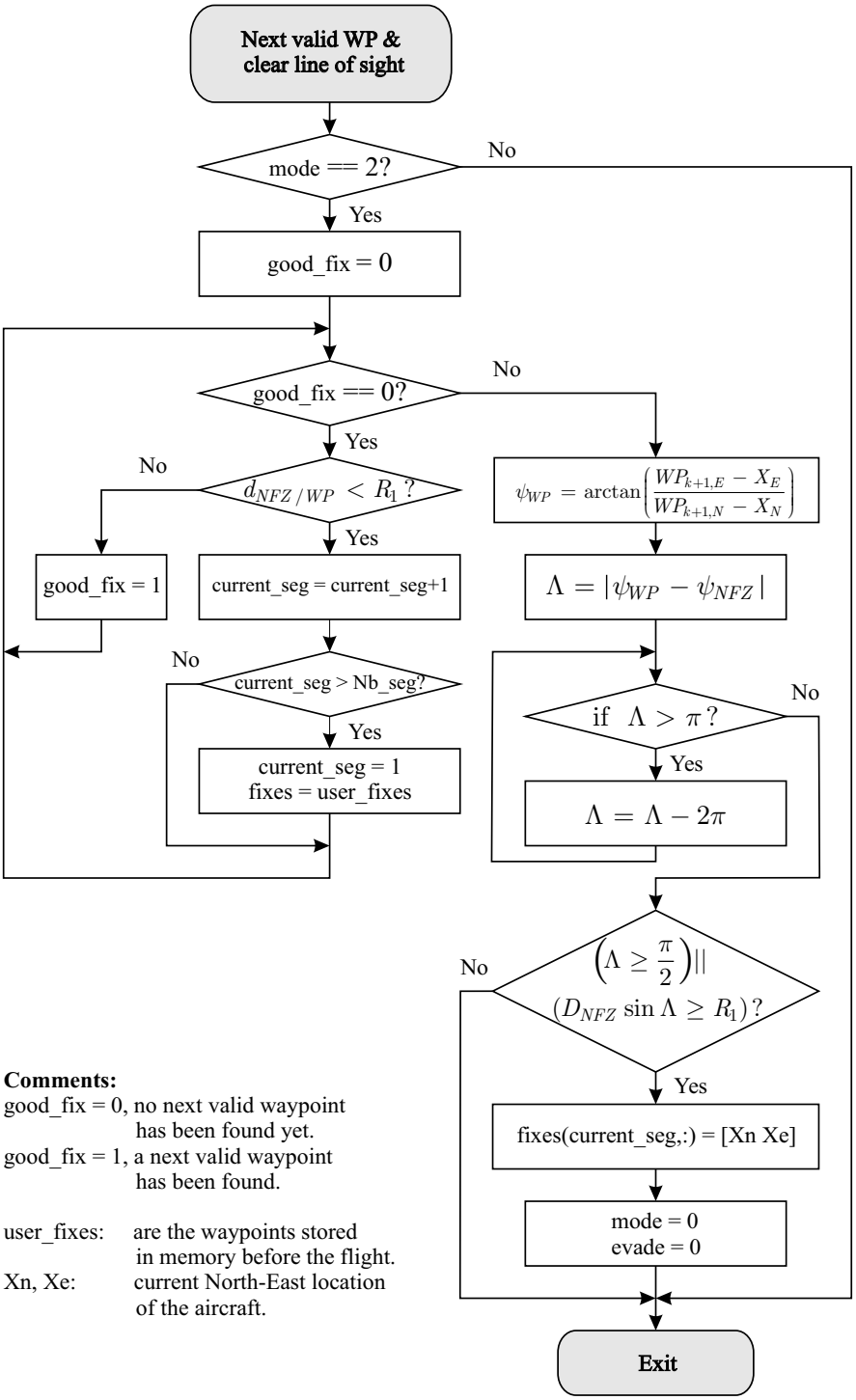


Fig. 9.18 Next valid point and clear line-of-sight determination

9.8.2 Simulation Results

9.8.2.1 No Wind

This first scenario, shown in Fig. 9.19, highlights the basic response of the aircraft to an NFZ blocking its path. The aircraft begins South of the NFZ and flies North along the desired path defined by the waypoints 1 to 5 and returns back to the runway. The desired path passes through an NFZ, but the aircraft deviates around it before returning to the desired path. The simulation was run at three different flight speeds, 15, 30, and 45 m/s. It can be seen that the aircraft begins its turn much later when flying at 15 m/s than when flying at 45 m/s. The airplane stays outside the NFZ at all three speeds.

9.8.2.2 With Wind

This second scenario, shown in Fig. 9.20, highlights the response of the aircraft in wind conditions. The desired path remains the same as in the first scenario. The path taken by the aircraft without wind and with wind are shown for comparison. The aircraft is flying at a nominal airspeed of 30 m/s.

A first flight is made with a 6 m/s crosswind blowing from West to East. In this case, the path followed by the aircraft is almost identical to the one without wind.

Another flight simulation is made with wind blowing from South to North with a speed of 6 m/s. The trajectory in the latter windy condition differs from the the nominal track (without wind) in the two turns that avoid the obstacle, where there is a maximum difference of 20 m. This is due to the fact that the wind speed adds to the airspeed resulting in a higher ground speed V_n than the one obtained with no wind. Thus, the evasion maneuver starts earlier when the aircraft first encounters the obstacle in between points 1 and 2. In between points 4 and 5, the wind is facing the aircraft, and therefore, the ground speed is lower than the one obtained with no wind. This explains why the evasion maneuver starts later.

In both cases, the NFZ is avoided. After the obstacle has been avoided, the guidance system resumes normal waypoint tracking. As expected from the lateral guidance law described in Sect. 9.2 there is no steady-state error when flying the circular arc around the NFZ as shown in Fig. 9.20.

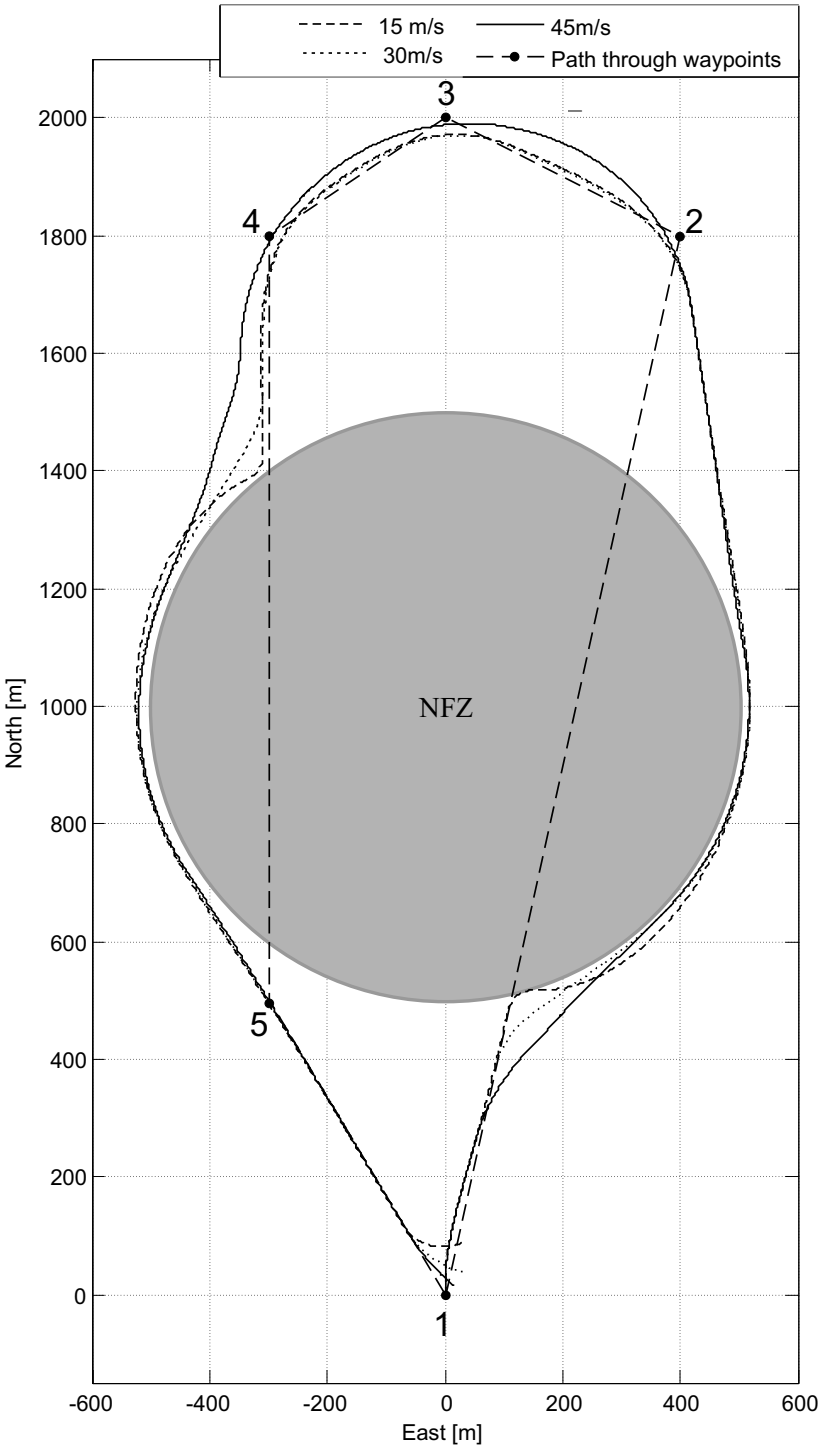


Fig. 9.19 Obstacle avoidance in no-wind condition at different speeds

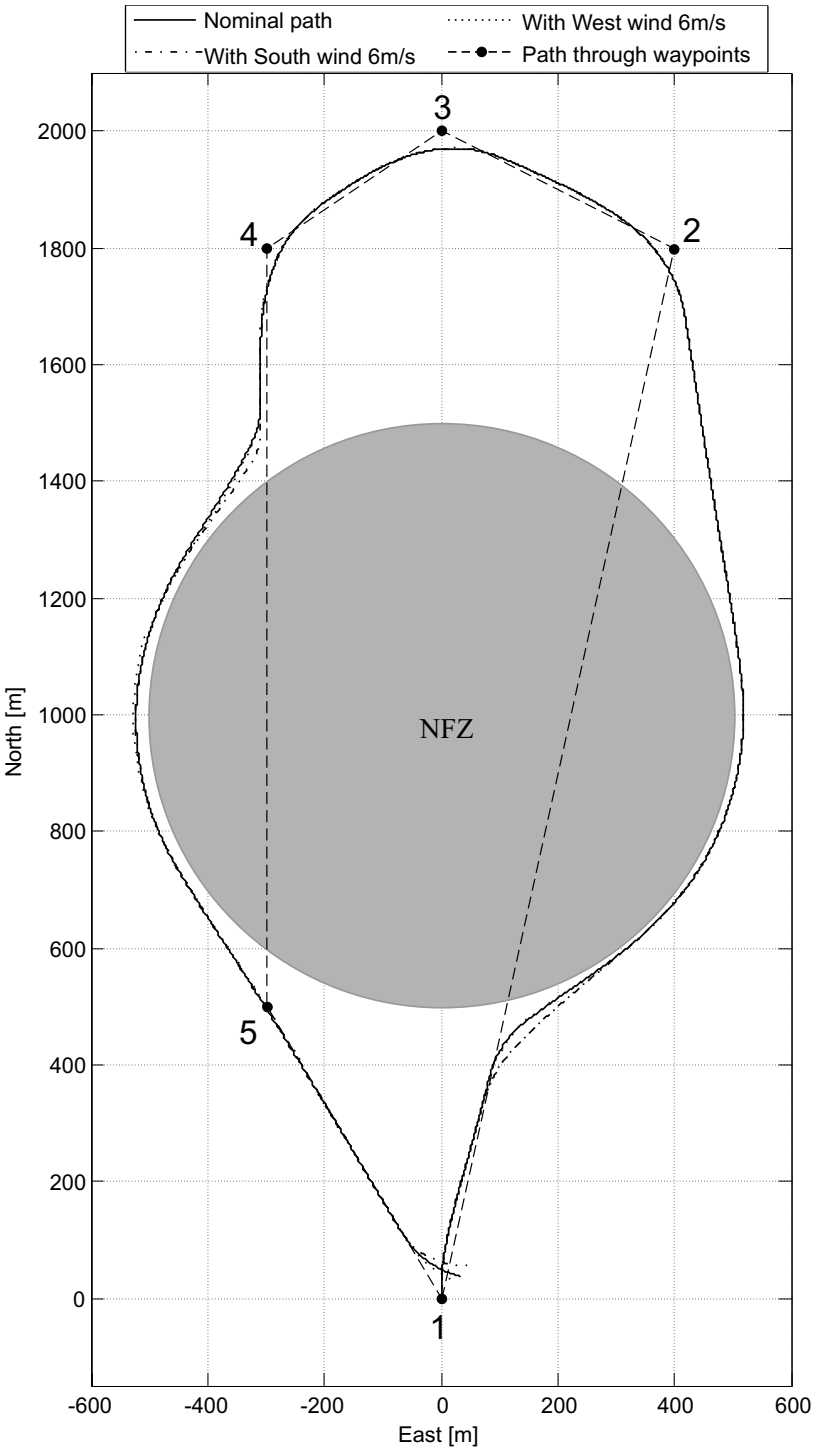


Fig. 9.20 Obstacle avoidance in wind conditions, $V_T = 30\,\mathrm{m/s}$

9.9 Conclusions

This chapter presented a guidance algorithm that combines simplicity of implementation and ability to avoid an NFZ. The algorithm successfully demonstrated in simulation its ability to guide the aircraft around the NFZ and then to resume flying along the desired path. The guidance system intrinsically takes into account the wind condition via the ground speed of the aircraft. Finally, the method is computationally efficient.

References

1. G. Ducard and H. P. Geering. A Computationally Efficient Guidance System for a Small UAV. In *Proceedings of the 4th International Conference on Informatics in Control, Automation and Robotics*, Angers, France, May 2007.
2. G. Ducard, K. C. Kulling, and H. P. Geering. A Simple and Adaptive On-Line Path Planning System for a UAV. In *Proceedings of the IEEE 15th Mediterranean Conference on Control and Automation*, pages 1–6, Athens, Greece, June 2007. T34-009.
3. L. Kavraki, P. Svestka, J. Latombe, and M. Overmars. Probabilistic Roadmaps for Path Planning in High-dimensional Configuration Spaces. *IEEE Transactions on Robotics and Automation*, 12(4):566–580, August 1996.
4. J. N. Amin, J. D. Boskovic, and R. K. Mehra. A Fast and Efficient Approach to Path Planning for Unmanned Vehicles. In *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, Keystone, CO, August 21−24 2006.
5. Y. Koren and J. Borenstein. Potential Fields Methods and their Inherent Limitations for Mobile Robot Navigation. In *Proceedings of IEEE Conference on Robotics and Automation*, Sacramento, CA, April 1991.
6. S. G. Loizou and K. J. Kyriakopoulos. Closed-loop Navigation for Multiple Holonomic Vehicles. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2861–2866, Minneapolis, Minnesota, 2002.
7. E. Rimon and D. Koditschek. Exact Robot Navigation Using Artificial Potential Functions. *IEEE Transactions on Robotics and Automation*, 8(5):501–518, October 1992.
8. Y. Kuwata, A. Richards, T. Schouwenaars, and J. P. How. Decentralized Robust Receding Horizon Control for Multi-vehicle Guidance. In *Proceedings of IEEE American Control Conference*, pages 2047–2052, Minneapolis, MN, June 2006.
9. S. Park. *Avionics and Control System Development for Mid-Air Rendezvous of Two Unmanned Aerial Vehicles*. PhD thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Available at http://hdl.handle.net/1721.1/16662, Cambridge, MA, 2004.
10. S. Park, J. Deyst, and J. P. How. A New Nonlinear Guidance Logic for Trajectory Tracking. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Providence, RI, 2004.